CLAIMS

1. A method for optimizing compiler-generated program code intended for a portable data carrier having both a processor core and a first and second memory area:

- the first memory area being provided to receive the optimized program code,
- the second memory area being provided to receive a predefined library having a multiplicity of library code fragments, and
- the compiler-generated program code being searched for program code fragments which correspond, at least in respect of their effect, to respectively one library code fragment, the program code fragments found thereby being replaced by respectively one call of the corresponding library code fragment.

2. The method according to claim 1, wherein a program code fragment is replaced by a library code fragment only if both code fragments are identical in their form as executable machine code.

3. The method according to claim 1, wherein at least some library code fragments are parameterised.

4. The method according to claim 1, wherein a program code fragment to be replaced is replaced, at least if the program code fragment does not interfere with the program flow, by a subroutine call instruction to the corresponding library code fragment.

5. The method according to claim 1, wherein the compiler-generated program code exists in the form of assembler source code, and the optimization procedure is performed on a source code level.

6. The method according to claim 1, wherein the predefined library is matched to at least one of the following:

- the hardware of the portable data carrier,
- an operating system of the portable data carrier, and
- a compiler used in the generation of the compiler-generated program code.

7. The method according to claim 1, wherein the first memory area is electrically programmable.

8. The method according to claim 1, wherein the second memory area is mask-programmable.

9. The method according to claim 1, wherein the first memory area occupies more chip area per memory cell in the portable data carrier than is occupied by the second memory area.

10. A computer program product comprising program instructions for a general-purpose computer which cause the general-purpose computer to optimize compiler-generated program code intended for a portable data carrier having both a processor core and a first and second memory area, the first memory area being provided to receive the optimized program code, the second memory area being provided to receive a predefined library having a multiplicity of library code fragments, wherein

the optimization includes searching the compiler-generated program code for program code fragments which correspond, at least in respect of their effect, to respectively one library code fragment, the program code fragments found thereby being replaced by respectively one call of the corresponding library code fragment.

11. The computer program product according to claim 10, wherein the program instructions additionally implement a compiler for converting a high-level language source code into the compiler-generated program code.

12. The computer program product according to claim 10, wherein a program code fragment is replaced by a library code fragment only if both code fragments are identical in their form as executable machine code.

13. The computer program product according to claim 10, wherein a program code fragment to be replaced is replaced, at least if the program code fragment does not interfere with the program flow, by a subroutine call instruction to the corresponding library code fragment.

14. The computer program product according to claim 10, wherein the compiler-generated program code exists in the form of assembler source code, and the optimization procedure is performed on a source code level.

15. The computer program product according to claim 10, wherein the predefined library is matched to at least one of the following:

- the hardware of the portable data carrier,
- an operating system of the portable data carrier, and

- a compiler used in the generation of the compiler-generated program code.

16. The computer program product according to claim 10, wherein the first memory area is electrically programmable, and the second memory area is mask-programmable, and the first memory area occupies more chip area per memory cell in the portable data carrier than is occupied by the second memory area.

17. A portable data carrier having a processor core, a first memory area and a second memory area, there being contained in the first memory area optimized program code, and there being contained in the second memory area a library which is predefined independently of the optimized program code and has a multiplicity of library code fragments, wherein the optimized program code has been obtained from compiler-generated program code by searching for program code fragments which correspond, at least in respect of their effect, to respectively one library code fragment, the program code fragments found thereby being replaced by respectively one call of the corresponding library code fragment.

18. The portable data carrier according to claim 17, wherein, when obtaining the optimized program code from the compiler-generated program code, a program code fragment is replaced by a library code fragment only if both code fragments are identical in their form as executable machine code.

19. The portable data carrier according to claim 17, wherein the predefined library is matched to at least one of the following:

- the hardware of the portable data carrier,
- an operating system of the portable data carrier, and
- a compiler used in the generation of the compiler-generated program code.

20. The portable data carrier according to claim 17, wherein the first memory area is electrically programmable, and the second memory area is mask-programmable, and the first memory area occupies more chip area per memory cell in the portable data carrier than is occupied by the second memory area.